



C# Avançado

Exercícios Propostos

Programação Multithread

1 Exercício

Crie uma aplicação onde 5 threads inserem 10 números randômicos em uma lista, a qual é compartilhada entre as threads. Quando todas terminarem de executar sua tarefa, imprima os números na tela.

É importante evitar que problemas de concorrência no acesso à lista ocorram, de forma que apenas uma thread deve poder acessar a lista por vez. Sincronize a menor porção de código possível, permitindo que haja concorrência nos locais onde ela pode realmente ocorrer.

Utilize um `CountdownEvent` para fazer com que a thread principal fique bloqueada aguardando todas as 5 threads realizarem suas tarefas antes da impressão dos números na tela.

2 Exercício

Implemente uma fila de execução de tarefas. Esta fila é uma thread que fica aguardando tarefas. Quando uma ou mais tarefas chegam, elas são executadas na ordem que chegaram.

Uma tarefa é representada por um `delegate Action`. Crie uma classe `QueueTask` para representar a fila de tarefas e um método `Enqueue()` que permite enfileirar uma tarefa para execução.

Utilize um `AutoResetEvent` para gerenciar a fila, de forma que a thread da fila de tarefas deve ficar bloqueada aguardando até que uma tarefa esteja disponível.

Escreva um código que enfileire algumas tarefas para testar o uso da fila.

3 Exercício

Na ciência da computação, existem alguns exemplo clássicos de problemas relacionados à programação concorrente. Um deles é o problema do "leitor e escritor".

Neste problema, existe uma série de leitores e escritores que agem sobre um mesmo conjunto de dados. Para evitar problemas de concorrência, os leitores podem ler os dados ao mesmo tempo, enquanto cada escritor deve ter acesso exclusivo aos dados no momento da escrita (não pode haver outro escritor escrevendo ou outro leitor lendo durante este processo). Um cenário prático deste problema é o acesso a informações em uma tabela de um banco de dados.

Crie uma aplicação que reproduz este cenário, onde múltiplas threads (leitoras e escritoras) ficam executando em loop simultaneamente, lendo e escrevendo dados de acordo com as regras mencionadas acima.

Para saber mais sobre este problema, você pode consultar os seguintes links (na internet existe bastante material a respeito também):

- <http://ces33.blogspot.com.br/2009/05/o-problema-do-leitores-e-escriitores-com.html>
- http://en.wikipedia.org/wiki/Readers-writers_problem